



UNITED STATES DEPARTMENT OF COMMERCE

Patent and Trademark Office

Address: COMMISSIONER OF PATENTS AND TRADEMARKS  
Washington, D.C. 20231

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.
-----------------	-------------	----------------------	---------------------

08/831,845 04/01/97

CALDER

B P2167/SUN1P1

HICKMAN BEYER & WEAVER  
PO BOX 61059  
PALO ALTO CA 94306

TM02/1206

EXAMINER

BULLOCK TR  
ART UNIT PAPER NUMBER

2151  
DATE MAILED:

12/06/00

Please find below and/or attached an Office communication concerning this application or proceeding.

Commissioner of Patents and Trademarks

BEST AVAILABLE COPY

<b>Office Action Summary</b>	Application No.	Applicant(s)
	08/831,845	CALDER ET AL.
	Examiner	Group Art Unit
	Lewis Bullock, Jr.	2151

Responsive to communication(s) filed on Sep 19, 2000

This action is **FINAL**.

Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11; 453 O.G. 213.

A shorter statutory period for response to this action is set to expire 3 month(s), or thirty days, whichever is longer, from the mailing date of this communication. Failure to respond within the period for response will cause the application to become abandoned. (35 U.S.C. § 133). Extensions of time may be obtained under the provisions of 37 CFR 1.136(a).

#### Disposition of Claims

Claim(s) 1-27 is/are pending in the application.

Of the above, claim(s) \_\_\_\_\_ is/are withdrawn from consideration.

Claim(s) \_\_\_\_\_ is/are allowed.

Claim(s) 1-27 is/are rejected.

Claim(s) \_\_\_\_\_ is/are objected to.

Claims \_\_\_\_\_ are subject to restriction or election requirement.

#### Application Papers

See the attached Notice of Draftsperson's Patent Drawing Review, PTO-948.

The drawing(s) filed on \_\_\_\_\_ is/are objected to by the Examiner.

The proposed drawing correction, filed on \_\_\_\_\_ is  approved  disapproved.

The specification is objected to by the Examiner.

The oath or declaration is objected to by the Examiner.

#### Priority under 35 U.S.C. § 119

Acknowledgement is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d).

All  Some\*  None of the CERTIFIED copies of the priority documents have been received.

received in Application No. (Series Code/Serial Number) \_\_\_\_\_.

received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

\*Certified copies not received: \_\_\_\_\_

Acknowledgement is made of a claim for domestic priority under 35 U.S.C. § 119(e).

#### Attachment(s)

Notice of References Cited, PTO-892

Information Disclosure Statement(s), PTO-1449, Paper No(s). \_\_\_\_\_

Interview Summary, PTO-413

Notice of Draftsperson's Patent Drawing Review, PTO-948

Notice of Informal Patent Application, PTO-152

--- SEE OFFICE ACTION ON THE FOLLOWING PAGES ---

Art Unit: 2151

## DETAILED ACTION

### *Claim Rejections - 35 USC § 103*

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

2. Claims 1, 2, 5, 6, 8, 9, 12-16, and 19-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over COLWELL (US 5,303,361).

As to claim 1, COLWELL teaches a computer-implemented framework for associating data (data files) with a command object (viewer module), the command object being arranged to operate on the data, wherein the data is associated with an application (user) (col. 4, lines 35-46), the computer-implemented framework comprising:

a data handler mechanism (user interface module) arranged to interface with the application (user) (col. 5, lines 23-25);  
a data retriever mechanism (index module) in communication with the data handler mechanism (user interface module), the data retriever mechanism being arranged to obtain the data (data files) and to pass the data to the data handler mechanism (col. 5, lines 25-34); and  
a mapping mechanism (viewer manager) in communication with the data handler mechanism (user interface module), the mapping mechanism being separate from the data handler mechanism (col. 8, lines 45-48), the mapping mechanism being arranged to obtain the command

Art Unit: 2151

object (viewer), wherein the command object (viewer) is obtained by the mapping mechanism (viewer manager) based substantially on the data (col. 8, line 53 - col. 9, line 17; col. 3, lines 58-62). It would be obvious that an user can be an invoking application.

As to claim 9, COLWELL teaches a computer-implemented method for associating data (data files) with a command object (viewer module) in response to a request from an application (user) (col. 4, lines 35-46), the method comprising:

accessing the data (data files) through an interface (user interface module) in response to the request from the application (user) (via the index module) (col. 4, lines 53-58; col. 5, lines 25-34), the interface being independent from the application (user) in communication with the application wherein the request from the application is processed by the interface (col. 3, lines 30-32);

accessing a mapping mechanism (viewer manager) which is in communication with the interface (user interface module), the mapping mechanism (viewer manager) being independent from the application (user) such that the mapping mechanism is not a component of the application (col. 8, lines 45-52), the mapping mechanism (viewer manager) being maintained separately from the interface (user interface module) (Fig. 1), the mapping mechanism further being arranged to locate a command object (viewer module) that is appropriate for the data (data file), wherein the mapping mechanism is accessed by the interface (user interface module) (col. 8, line 53 - col. 9, line 17; col. 3, lines 58-62);

Art Unit: 2151

obtaining the command object (viewer module) that is appropriate for the data (data file), wherein the mapping mechanism obtains the command object and passes the obtained command object to the interface (col. 8, line 53 - col. 9, line 17; col. 3, lines 58-62);

binding the command object (viewer module) to the data (data file), where the interface binds the command object to the data (via bid) (col. 8, line 61-col. 9, line 2); and

returning the command object to the application, wherein the interface returns the command object to the application (col. 9, lines 13-16). It can be obvious that the user is an invoking application.

As to claim 2, COLWELL teaches the data (data file) is a stream of bytes (reads the first 1,000 bytes) (col. 8, lines 56-60), and the data handler mechanism is arranged to bind the stream of bytes to the command object (col. 5, lines 23-49).

As to claim 5, COLWELL teaches the data is one of text data and image data (col. 3, lines 62-65).

As to claim 6, COLWELL teaches the data handler (user interface module) is further arranged to receive a request from the application (user), bind the data to the command object (viewer) (via bid), and to return the command object to the application (col. 5, lines 23-49).

Art Unit: 2151

As to claim 8, It would be obvious that since viewer modules bid on data files to see which one processes the file (col 8, line 53 - col. 9, line 17) that the viewer manager can have a table to determine which viewer module can process the file (col. 8, lines 61-68).

As to claim 21, COLWELL teaches the command object (viewer module) is obtained by the mapping mechanism (viewer manager) based substantially on the data (portion of the data file) without an external input from a user of the application (col. 3, lines 57-62).

As to claim 22, COLWELL teaches the command object (viewer module) is obtained by the mapping mechanism (viewer manager) based substantially on the data (portion of the data file) without directly involving the application (col. 3, lines 57-62).

As to claim 26, COLWELL teaches the mapping mechanism (viewer manager) and the data handler mechanism (user interface module) are separately maintained (fig. 1).

As to claim 12, COLWELL teaches accessing a data retriever (index module) which is arranged to obtain the data (data file), wherein the data is a stream of bytes (reads the first 1,000 bytes) (col. 8, lines 56-60).

Art Unit: 2151

As to claim 13, COLWELL teaches operating on the data (data file) using the command object (viewer module) (col. 3, lines 58-62).

As to claim 14, COLWELL teaches the command object (viewer module) that is appropriate for the data (data file) is selected from a set of command objects (viewer modules) (col. 3, lines 58-68). It would be obvious that the viewer modules can be stored in a list and accessed.

As to claim 15, COLWELL teaches receiving a request from the application, the request being received by the interface, wherein the interface performs the steps of: obtaining a type associated with the data (read first portion); obtaining the command list (viewer modules) through the mapping (viewer manager); and returning the command list (viewer modules) to the application (user) (col. 3, lines 58-68; col. 8, line 53 - col. 9, line 17). It would be obvious that since a plurality of viewer modules can be accessed that they can be stored in a list and accessed.

As to claims 16, 19, and 20, reference is made to a computer program product which corresponds to the method of claims 9, 13, and 14 and is therefore met by the rejection of claims 9, 13, and 14 above. Claim 16 also details the mapping mechanism is not a part of the application. It would be obvious that since COLWELL teaches the mapping mechanism is not a

Art Unit: 2151

component of the application as detailed in claim 9 that is also not a part of the application as claimed.

As to claims 23 and 24, refer to claims 1 and 26 for rejection. However, claim 23 further details the data handler mechanism is independent and interfacing with a plurality of applications. COLWELL teaches the data handler mechanism (user interface module) is independent (col. 3, lines 30-32). It would be obvious that since the interface module is part of the search and retrieval system and takes request from a users (col. 3, lines 30-32), then it can take request from a plurality of users.

As to claim 25, COLWELL teaches the mapping mechanism (viewer manager) is not a component of the data handler mechanism (user interface module) (fig. 1).

As to claim 27, It would be obvious that since the viewer modules are independent of the interface user module and can be dynamically added without change the interface module (col 8, lines 38-52) then the mapping mechanism (viewer manager) is not specific to the application (user) while the data handler mechanism (user interface module) is specific to the application (directly connected to user) (col. 3, lines 30-32).

Art Unit: 2151

3. Claims 3, 7, 10, and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over COLWELL (US 5,303,361) in view of PAYNE (US 6,021,433).

As to claim 3, COLWELL teaches the data handler mechanism (user interface module) is further arranged to bind the data object to the command object (col. 8, line 53-col. 9, line 17). However, COLWELL does not teach the data content handler mechanism. PAYNE teaches a data content handler mechanism (content manager of the central broadcast server) in communication with the data handler mechanism (message server design) (col. 6, lines 28-42), the data content handler mechanism being arranged to convert the data into a data object (col. 8, lines 26-47). Therefore, it would be obvious to combine the teachings of COLWELL with the teachings of PAYNE in order to dynamically access data (col. 2, lines 42-64).

As to claim 7, COLWELL teaches the data handler mechanism (user interface module) is arranged to bind the data object (data file) to the command object (viewer module) (col. 8, line 53 - col. 9, line 17). However, COLWELL does not teach the data content mechanism or the data source mechanism. PAYNE teaches a data source mechanism (information sources / servers) arranged to obtain a stream of bytes (col. 7, lines 43-56) and a data content handler mechanism (content manager) arranged to convert the stream of bytes into a data object (col. 8, lines 26-47), the data source mechanism being in communication with the data content handler mechanism (fig. 2). Refer to claim 3 for the motivation to combine.

Art Unit: 2151

As to claim 10, COLWELL substantially disclose the invention. However, COLWELL does not teach the data content handler mechanism. PAYNE teaches: passing a stream of bytes (information) to a data content handler mechanism (content manager) arranged to create a data object (compressed) from the stream of bytes; and passing the data object to the interface (message server design) (col. 6, lines 28-42), wherein the data is the data object (compressed).

Refer to claim 3 for the motivation to combine.

As to claim 17, reference is made to a computer program product which corresponds to the method of claim 10 and is therefore met by the rejection of claim 10 above.

4. Claims 4, 11, and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over COLWELL in view of PAYNE as applied to claim 3 above, and further in view of "The Java Language Environment" by SUN.

As to claim 4, COLWELL and PAYNE substantially disclose the invention above. However, neither reference teaches the cited functionality. SUN teaches the data object (object) is created using the Java programming language, and the command object is a Java command object (Java code to support object) (pg. 77-78). Therefore, it would be obvious to combine the teachings of COLWELL with the teachings of PAYNE and SUN in order to dynamically perform or add capabilities (pg. 75-76, 72).

Art Unit: 2151

As to claim 11, COLWELL and PAYNE substantially disclose the invention above.

However, neither teach the cited functionality. SUN teaches the data object (object) is created using the Java programming language, and the command object is a Java command object (Java code to support object) (pg. 77-78). Refer to claim 4 for the motivation to combine.

As to claim 18, reference is made to a computer program product which corresponds to the method of claim 11 and is therefore met by the rejection of claim 11 above.

***Response to Arguments***

5. Applicant's arguments with respect to claims 1-26 have been considered but are moot in view of the new ground(s) of rejection.

6. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37

Art Unit: 2151

CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Lewis A. Bullock, Jr. whose telephone number is (703) 305-0439.

  
MAJID A. BANANKHAH  
PRIMARY EXAMINER

lab

November 28, 2000